

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Řízení robota LEGO MINDSTORMS NXT v prostředí LabVIEW

Control of robot LEGO MINDSTORMS NXT in development system LabVIEW

BAKALÁRSKÁ PRÁCE

Jiří Kučera

Studijní program: B2612 – Elektrotechnika a informatika
Studijní obor: 2612R001 - Elektronické informační a řídicí systémy
Pracoviště: Ústav řízení systémů a spolehlivosti
Fakulta mechatroniky a mezioborových inženýrských studií
Technická Univerzita v Liberci
Studentská 2, 461 17, Liberec 1

Vedoucí práce: Ing. Jiří Bažant
Konzultant: Ing. Jaroslav Vlach, PRECIOSA, a.s. Jablonec n. N.
Datum a místo vydání: 12. 5. 2008 v Liberci

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že Technická univerzita v Liberci má právo na uzavření licenční smlouvy o užití mé bakalářské práce a prohlašuji, že souhlasím s případným užitím mé bakalářské práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své bakalářské práce či poskytnout licenci k jejímu využití mohu jen se souhlasem Technické univerzity v Liberci, která má právo požadovat ode mne přiměřený příspěvek na úhradu nákladů vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury pod vedením školitele.

V Liberci 12. 5. 2008

.....
Jiří Kučera

Poděkování

Chtěl bych poděkovat všem, kteří mi pomáhali při řešení bakalářské práce. Firmě PRECIOSA a.s. za zapůjčení technického vybavení Zaměstnancům firmy PRECIOSA a.s. zejména panu ing. Jaroslavu Vlachovi za podmětné konzultace při řešení problémů a panu Josefu Havlíčkovi za technickou spolupráci.

Poděkování patří též panu ing. Jiřímu Bažantovi, vedoucímu bakalářské práce při TUL, za vedení bakalářské práce.

Abstrakt

Práce je teoreticky zaměřena na prozkoumání možností robota od firmy LEGO a využitím robota pro průmyslové aplikace. Práce se zabývá porovnáním základního programovacího prostředí LEGO MINDSTORMS Education NXT se standardně používaným průmyslovým grafickým vývojovým prostředím LabVIEW. Dále se práce prakticky zabývá tvorbou jednotlivých aplikací zkoumajících funkce robota ve zmíněných programech a možností použití robota pro simulaci průmyslových aplikací.

Klíčová slova: LabVIEW, LEGO MINDSTORMS NXT, robot, LEGO MINDSTORMS Education NXT, řízení

Abstract

This work is abstractedly bent on searching occasion of robot by LEGO Company and using of robot for industrial application. The work deal with comparison basic programming setting LEGO MINDSTORMS Education NXT one another standart use by industrial graphic development system LabVIEW. Works further practically deal by creation of single application examining robot function in mentioned programs and possibility of using robot for simulation of industrial application.

Key words: LabVIEW, LEGO MINDSTORMS NXT, robot, LEGO MINDSTORMS Education NXT, control

Obsah

Úvod.....	8
1 Cíle práce	9
Teoretická část	10
2 Popis základních částí LEGO MINDSTORMS NXT	10
2.1 Cíle teoretické části práce	10
2.2 Základní jednotka (kostka NXT)	10
2.2.1 Procesory	11
2.2.2 vstupní porty	12
2.2.3 výstupní porty.....	13
2.2.4 Displej	14
2.2.5 Zvuk	15
2.2.6 Bluetooth komunikace	15
2.2.7 Firmware	17
2.3 Periferie	18
2.3.1 Vstupní periferie.....	18
2.3.2 Výstupní periferie.....	21
2.3.3 Doplnkové díly.....	23
2.4 LEGO MINDSTORMS Education NXT	23
2.4.1 Programová část.....	24
2.4.2 Robotc	28
3 LabVIEW	30
3.1 Popis prostředí LabVIEW	30
3.1.1 Front Panel	31
3.1.2 Block Diagram	33
3.2 LabVIEW toolkit for LEGO MINDSTORMS NXT.....	35
3.2.1 NXT Direct Commands	35
3.2.2 NXT Toolkit.....	36
4 Praktická část	38
4.1 Cíle praktické části práce	38
4.2 Konstrukce robota	38
4.3 Zadání úlohy	38
4.3.1 Objíždění překážky	39
4.3.2 regulace motorů v závislosti na vzdálenosti a aktuálním výkonu motorů.....	40

4.3.3 Ostatní úlohy	41
4.3.4 Řešené problémy	41
5 LabVIEW x LEGO MINDSTORMS Education NXT	43
5.1 LEGO MINDSTORMS Education NXT	43
5.2 LabVIEW	44
6 Závěr	45
Příloha A	47
Příloha B	48

Úvod

Slovo robot znamená samostatně pracující mechanický stroj vykonávající určené úkony. Prvně slovo robot použil v roce 1920 Karel Čapek ve své divadelní hře RUR (Rossum's Universal Robots), i když vynálezcem tohoto slova byl jeho bratr Josef Čapek. Pojem slova robot zastává v dnešním světě mnoho různých podob, jako jsou průmyslové roboty, kuchyňský robot nebo také droid, kyborg atd. Všechny tyto pojmy však sdružuje pojem, proč jsou vlastně vyrobeny. Prvotní důvod k tomuto počínání je snaha ulehčit lidem práci nebo tuto práci zpřesnit. Proto jsou roboty implementovány do různých lidských činností od výroby přes transport až po roboty pomáhající chirurgovi při operaci. Podstatnou výhodou těchto robotů je také fakt, že nepotřebují odpočinek, nikdy nebudou stávkovat. Avšak jsou zde i rizika s provozem robota. Robot pracuje podle daného algoritmu, v přesně dané oblasti a má dostatečně tuhou konstrukci. Tyto části se však mohou poškodit a způsobit tak mnoho vážných havárií. Představa, že by zdravotní robot při chirurgické operaci přestal vykonávat úkony zadané operátorem, je nepředstavitelná. Proto při konstrukci a návrhu programového vybavení robota se musí velmi dbát na možné chyby.

V práci se budou řešit konstrukční prvky robota, jeho programové vybavení, řízení robota jako autonomní systém a přímé řízení robota přes počítač. Samotná práce bude rozdělena na část teoretickou a část praktickou.

Teoretická část se bude zabývat obecným popisem LEGO MINDSTORMS NXT jeho jednotlivými technickými komponentami, periferiemi, firmwarem a jeho vazbou na starší modely a alternativní přípojně modulů jako jsou senzory. Dále v teoretické části bude popisováno programové prostředí LEGO MINDSTORMS Education NXT, tvorba programu v tomto prostředí, popisování programu a jeho možností ovládání a komunikace s robotem. Další část teoretické rozpravy se bude zabývat grafickým vývojovým prostředím LabVIEW, jeho nástroji, popisem grafického prostředí a ukázkou návrhu jednoduchých programů jakožto i popisem LabVIEW toolkitu for LEGO MINDSTORMS NXT a jeho ovládání robota.

Praktická část práce se bude zabývat popisem jednotlivých programů vytvořených v programu LabVIEW pomocí LabVIEW toolkitu for LEGO MINDSTORMS NXT. Programy budou navrženy jako autonomní systémy řešící základní problémy robota při jeho pohybu a rozhodování, nebo jako roboty s přímým řízením z počítače. V další části praktické rozpravy budou srovnány výhody a nevýhody tvorby programu pro robota v prostředí LEGO MINDSTORMS Education NXT a v prostředí LabVIEW.

1 Cíle práce

Cílem bakalářské práce je prostudování vlastností a možností robota LEGO MINDSTORMS NXT, jeho konstrukčních prvků, jednotlivých funkčních komponent, řízení robota, možností simulace a řízení technologických aplikací. Dalším cílem je seznámení se s programovým prostředím LEGO MINDSTORMS Education NXT, které je určeno přímo pro řízení tohoto robota. Seznámení se s grafickým vývojovým prostředím LabVIEW a jeho nástroji, vytvoření demonstračního programu pro modelové odzkoušení možností robota a srovnání programovacího prostředí LabVIEW s programovacím prostředím LEGO MINDSTORMS Education NXT na základě návrhu aplikací pro řízení robota LEGO. Cílem je též zhodnocení dosažených výsledků a možnostmi použití robota jako modelu pro průmyslové aplikace v rozmezí od použitého softwaru až po konstrukční možnosti robota.

Teoretická část

2 Popis základních částí LEGO MINDSTORMS NXT

2.1 Cíle teoretické části práce

Teoretická část má za cíl seznámit se s konstrukčními prvky robota. Prozkoumat vlastnosti a možnosti programovacího prostředí LabVIEW, jeho nástroje a programovací prostředí LEGO MINDSTORMS Education NXT. Seznámení se s prostředím Robotc a LabVIEW toolkit for LEGO MINDSTORMS NXT.

2.2 Základní jednotka (kostka NXT)

Řídící jednotka „NXT Kostka” tvoří základní část stavebnice LEGO MINDSTORMS NXT. NXT kostka je postavena na dvou mikroprocesorech ATMEEL. Tyto dva procesory tvoří základní část řídicího mechanismu NXT kostky. Samotná řídicí jednotka má tvar kvádrů. Na své spodní straně je překryta šedým krytem, který zakrývá všechny periferní konektory a na svrchu je překryta bílým krytem, který obepíná displej a tlačítka. NXT kostka má velikost 11,2 x 7,2 x 4,6 cm (délka x šířka x hloubka), což odpovídá 14 x 9 x 6 spojovacích LEGO bodů. Celý pohled na NXT kostku nám přiblíží **obr. 2. 1**. NXT kostka je na boku vybavena čepy pro přichycení ostatních dílů LEGO TECHNIK. Na každé straně je 10 čepů uspořádaných do dvou polí ve tvaru písmena L. Čepy jsou rovněž na spodní straně kostky. Na spodní straně jich je 12. Jsou rozděleny na dvě části po šesti, kde každá šestice má tvar přímky. Na kostce se nacházejí 4 vstupní porty (připojení senzorů), 3 výstupní porty (připojení motorů a led diod), 1 USB konektor (připojení kostky k počítači). Na kostce se nachází konektor pro napájení kostky přes adaptér ze sítě, pod nímž jsou umístěny dvě diody označující stav, ve kterém se kostka aktuálně nachází (červená LED dioda označuje stav nabíjení baterie, zelená LED dioda označuje stav napájení ze sítě). Pod krytem na spodní straně je umístěn akumulátor, který napájí motory, senzory atd., nebo zde může být umístěno 6 ekvivalentních 1,5V AA baterií. Na svrchní straně je umístěn displej, pod kterým jsou umístěna 4 tlačítka, kterými je možno ovládat firmware nebo jednotlivé programy uložené v kostce (spuštění a zastavení programu, vymazání...).

Na boku NXT kostky se nachází reproduktor.



obr. 2.1 - kostka NXT 1

Komponenty NXT kostky:

2.2.1 Procesory

V architektuře NXT kostky se jedná o základní stavební kameny. NXT kostka obsahuje dva mikroprocesory ATMEL.

ARM procesor

32 bitový mikroprocesor AT91SAM7S256 založený na jádru typu RISC. Obsahuje 256 KB flash paměť, 64 KB SRAM a pracuje na frekvenci 48 MHz. Ve struktuře jednotky zastává funkci hlavního procesoru

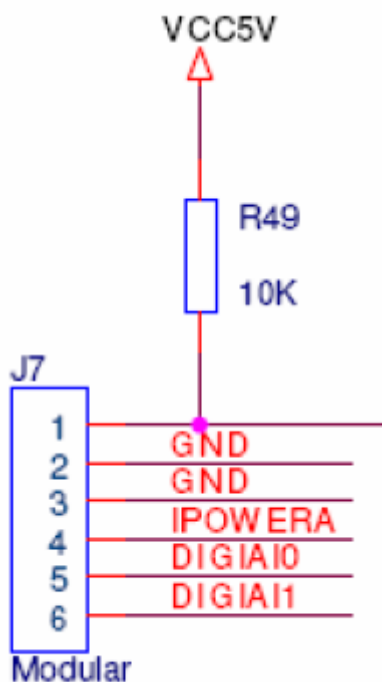
AVR procesor

8. bitový mikroprocesor ATmega48 založený na jádře RISC. Obsahuje 4 KB flash paměť, 512 B RAM paměť a pracuje na frekvenci 8 MHz. Ve struktuře jednotky zastává funkci koprocessoru.

K těmto procesorům jsou přes sběrnice přivedeny další části, jako jsou vstupy, výstupy, USB atd.

2.2.2 vstupní porty

LEGO MINDSTORMS NXT používá 4 vstupní porty, na které může být připojen jakýkoliv senzor (jednotlivé senzory mohou mít rozdílné parametry fyzického slova). Každý konektor má digitální i analogové rozhraní. Na **obr. 2.2** je patrné zapojení konektoru RJ12 (zapojení je identické pro všechny 4 vstupní porty).



obr. 2.2 - zapojení vstupního portu 1

Pin 1, ANA	analogový vstup a aktuální výstupní signál
Pin 2, GND	země
Pin 3, GND	země
Pin 4, IPOWERA	výstupní napájení 4,3 V (napájení senzorů)
Pin 5, DIGIAI0	digitální I/O pin propojená s procesorem ARM7
Pin 6, DIGIAI1	digitální I/O pin propojená s procesorem ARM7

Pin 1 (ANA) je analogový vstupní pin, který spojuje vstup s 10 - bitovým A/D převodníkem uvnitř procesoru AVR. Je vzorkovaný se stejnou vzorkovací frekvencí pro všechny analogové senzory a to s frekvencí 333MHz.

Pin 4(IPOWERA) Tento pin je vnitřně spojen se všemi vstupy i výstupy a je koncipován na aktuální výkon max. 180 mA. To znamená, že každý port má maximálně dostupný výkon přibližně 20 mA. Pokud je odebírán větší výkon, sníží se napájení automaticky. Je-li signál zkratovaný, NXT kostka se resetuje.

Digitální I/O piny (DIGIAI0, DIGIAI1) využívají komunikaci I^2C s rychlostí 9600 bit/s. NXT může být pouze master v I^2C komunikaci.

Vysokorychlostní komunikační port

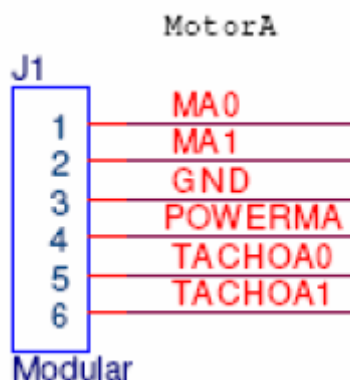
Vstupní port 4 funguje u NXT kostky jako vysokorychlostní komunikační port. Jedná se o RS485, kde na tento vstup se dá připojit více zařízení. I když není dodáván ke stavebnici žádný senzor, který by využíval tuto rychlost, do budoucna se počítá, že by se ke kostce mohlo takovéto zařízení připojit.

I^2C komunikace

V kostce mají všechna digitální rozhraní implementovanu I^2C komunikaci. Tato komunikace byla vynalezena firmou Philips v průběhu osmdesátých let. Jedná se o komunikaci mezi procesorem a externím zařízením připojeným na vstup. Po I^2C sběrnici komunikují i oba procesory. Kostka má vždy pro jeden vstupní port 4 komunikační kanály. Procesor kontroluje datový tok v každém kanálu a funguje jako master, ostatní zařízení připojená na porty fungují jako slave zařízení.

2.2.3 výstupní porty

LEGO MINDSTORMS NXT používá 3 výstupní porty pro ovládání jak motorů, tak přes redukci led diod. Konektory jsou zde použity stejné jako u vstupních portů a to konektor RJ12. Porty jsou realizovány tak, že výstupní zařízení může posílat informace zpět do NXT kostky, aniž by potřebovalo vstupní port. Všechny tři porty jsou identické (**obr. 2.3**).



obr. 2.3 - zapojení výstupního portu 1

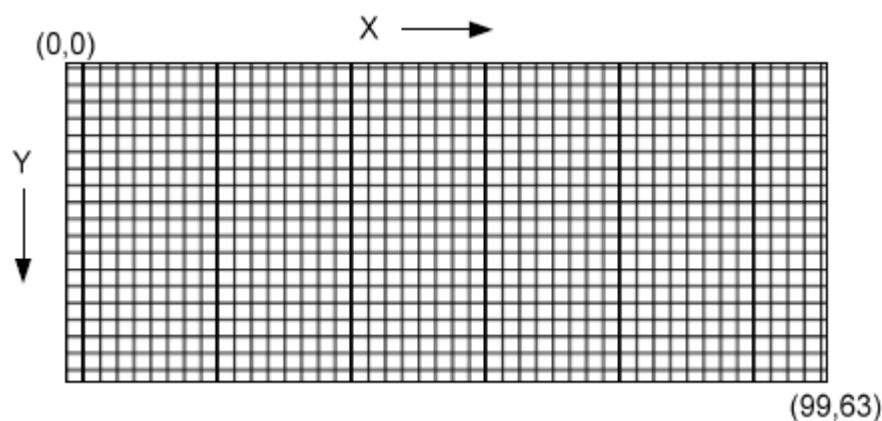
Pin 1, MA0	výstupní signál PWM pro akční členy
Pin 2, MA1	výstupní signál PWM pro akční členy
Pin 3, GND	země
Pin 4, POWERMA	výstupní napájení 4,3 V (napájení motorů a diod)
Pin 5, TACHOA0	vstupní hodnota zahrnující Schmittův klopný obvod
Pin 6, TACHOA0	vstupní hodnota zahrnující Schmittův klopný obvod

Piny MA0 a MA1 kontrolují akční členy (motory). Pro všechny motory je zaručen příkon 700 mA, ve špičce však maximálně 1 A. Výstupní PWM signál může kontrolovat brzdění. Řízení motoru má zabudovanou tepelnou ochranu, která při nepřetržitém chodu automaticky nastaví výstupní proud.

TACHOA0 a TACHOA1 jsou vstupní piny, které nám počítají počet pulsů jdoucích z motoru nebo určují, zda motor běží po směru nebo v protisměru.

2.2.4 Displej

Jedná se černobílý grafický bodový maticový displej s rozlišením 100 x 64 pixelů což odpovídá 20 x 40,6 mm. K ovládání displeje se používá řadič Ultra Chip 1601. Displej je spojen s mikroprocesorem ARM přes sběrnici SPI. Řádky displeje se aktualizují každých 17 ms. Data vstupující do displeje jsou nejdříve uložena do dvojdimenzionálního pole ve tvaru Normal[8][100],(Normal[Y/8][X]). Data jsou posílána do LCD řadiče následovně: 1. bajt ovládá prvních 8 pixelů svisle a druhý bajt ovládá 8 pixelů vodorovně (oba bajty začínají na (0,0)). Viz (obr. 2.4).



obr. 2.4 - místění pixelů v displeji 1

2.2.5 Zvuk

NXT kostka obsahuje čip zesilující zvuk pro zvýšení úrovně kvality zvuku. Zvukový výstup je ovládán mikroprocesorem ARM7 přes PWM. Reprodukter zabudovaný uvnitř kostky má impedanci 16 Ohm a průměr 21 mm. Při různých frekvencích má reproduktor různý odběr (**tab. 2.1**).

Frequency	Current mA	Power mW
440 Hz	102	169
4 KHz	78	97

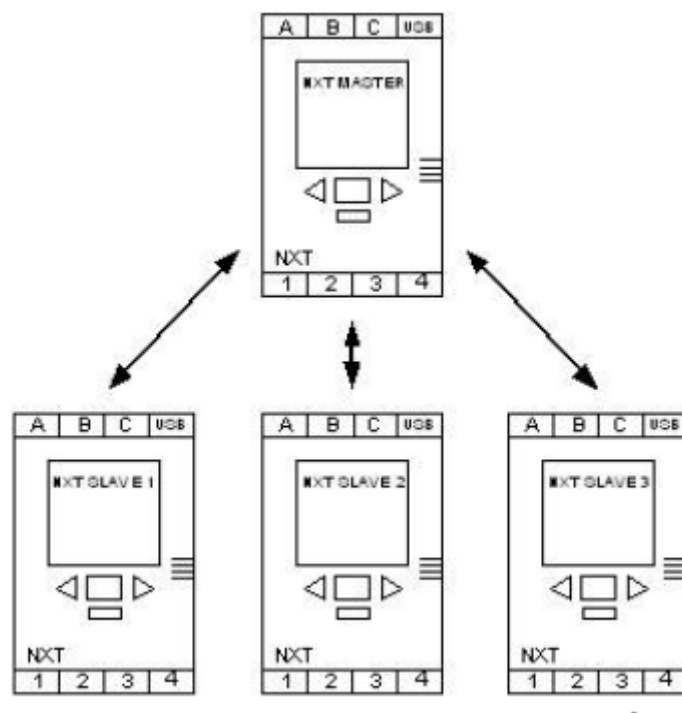
Tab. 2.1- aktuální spotřeba reproduktoru

Komunikaci s robotem LEGO MINDSTORMS NXT zabezpečuje komunikace přes USB nebo bezdrátová technologie Bluetooth.

2.2.6 Bluetooth komunikace

NXT kostka může být spojena až se třemi různými zařízeními, avšak jen s jedním ve stejný čas. Tohoto principu je využito v SPP (Serial port profile). NXT kostka komunikuje s bluetooth zařízením, které podporuje jednak LEGO MINDSTORMS NXT communication protokol a také podporuje SPP. V komunikaci mezi bluetooth zařízením a kostkou je možné posílat program a při vykonávání tohoto programu přijímat data. Maximální vzdálenost, na kterou lze posílat data mezi bluetooth zařízením a NXT kostkou, je 10m. Komunikace je nastavena jako master/ slave komunikace, tzn., že jedna NXT kostka je vždy master a ostatní

musí být slave. Při komunikaci mezi těmito kostkami jde přes kostku master veškerá komunikace (**obr. 2.5**).



obr. 2.5 - bluetooth komunikace 1

Pokud začne např. slave 1 posílat data, master je v daný okamžik přijímá. Jsou-li v době přijímání dat mastrem posílána data i z jiných slave zařízení, jsou nejprve vyhodnocena data ze slave 1, následně se master přepne, a jsou přijímána data z ostatních slave zařízení. Pro bluetooth komunikaci se používají 4 kanály. Kanál 0 se používá vždy pro komunikaci mezi slave NXT zařízením a master NXT zařízením (směrem k master NXT zařízením), jen tehdy, pokud jsou kanály 1, 2, 3 použity k odcházející komunikaci mezi Master NXT zařízením a slave NXT zařízením.

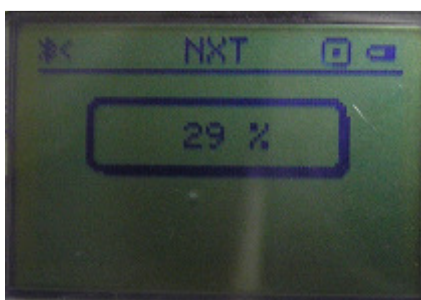
Kostku lze též připojit přes rozhraní USB 2.0. Přes USB lze do kostky nahrávat program i z kostky přijímat hodnoty od senzorů a motorů. Přes USB lze komunikovat rychlostí 12 MB/s. Kostka je vybavena čtyřmi tlačítky. Tlačítka jsou poskládána na svrchní straně do tvaru písmena T, viz. (**obr. 2.6**). Oranžové tlačítko nám slouží pro vypínání, zapínání a potvrzování úloh. Další dvě tlačítka umístěná po levé a pravé straně slouží pro pohyb v menu. Tmavě šedé tlačítko umístěné pod oranžovým tlačítkem nám umožňuje vracení se v menu, nebo se používá pro vymazání informace. Veškeré zapojení jak procesorů, tak jednotlivých částí NXT kostky najdete v příloze (**příloha A**).



obr. 2.6 - tlačítka 1

2.2.7 Firmware

Jedná se o program uložený v paměti NXT kostky. Každý program, který chce komunikovat s NXT kostkou, musí mít přesně danou verzi firmwaru. Jednotlivé firmwary mohou využívat jiným způsobem procesory uložené v kostce, nebo mohou mít funkce pro lepší práci například se senzory. Firmware slouží jako uživatelské rozhraní pro zprávu v paměti uložených programů, testování senzorů atd. Při spuštění NXT kostky se na displeji objeví uživatelské rozhraní firmwaru. Pod horním okrajem displeje je vidět, jakým typem připojení jsme aktuálně připojeni, jméno kostky, je-li kostka v provozu a stav baterií. Tyto systémové informace jsou uvedeny v každém submenu, s výjimkou programů, které přímo zapisují na displej a využívají celou jeho plochu. Pod touto informační lištou se nalézá úvodní menu. V tomto menu můžeme plně využívat všech vlastností firmware potažmo NXT kostky. Můžeme zde zkusit jednotlivé motory, senzory atd. Jejich informace vidíme v procentuálním vyjádření. (obr. 2.7)



obr. 2.7 - procentuální vyjádření 1

Dále zde jsou informace o komunikaci bluetooth, můžeme zde nastavit jednotlivé periferie nebo jsou zde uloženy jednotlivé zvukové doprovody, informace o vstupech a výstupech.

2.3 Periferie

Periferie můžeme rozdělit na vstupní a výstupní. Do vstupních periferií zahrnujeme senzory a mezi výstupní periferie pak motory, blikače a reproduktor. Všechny díly jsou navzájem zaměnitelné. Tudíž je můžeme libovolně kombinovat s ostatními díly. Jednotlivé periferie mají obdobný design, proto nenarušuje žádná část celkový vzhled robota. Jednotlivé periferie mají několik čepů pro přichycení na nosnou konstrukci a dávají tím robotu jedinečnou kreativitu návrhu.

2.3.1 Vstupní periferie

Do vstupních periferií se zahrnují následující senzory: ultrasonický senzor, dotkový senzor, světelný senzor, zvukový senzor. Tyto senzory dávají robotovi informace o okolním světě, jsou jeho očima a ušima. Robot jejich prostřednictvím může snímat informace o okolním světě, jako jsou vzdálenost od překážky, barva překážky, nebo intenzita zvuku vydávaného nějakým tělesem.

Ultrasonický senzor

Jedná se o senzor, který dává robotovi zrak. Robot, díky tomuto senzoru může vidět objekty, vyhnout se překážce nebo jen detekovat pohyb. Senzor má přijímací a vysílací část a pracuje na podobném principu jako echolokátor. Pošle zvuk a počítá, za jakou dobu se zvuk odrazí a vrátí zpět. Tento čas následně převede na vzdálenost. Ultrasonický senzor může vzdálenost měřit buď v cm, nebo v palcích. Snímá objekty 0 cm vzdálené až po objekty vzdálené 255 cm s přesností ± 3 cm. Senzor nejlépe detekuje předměty velké, avšak u předmětů, které jsou malé, tenké, nebo dosti zakřivené se může stát, že je senzor nezachytí. Poměrně velkým problémem jsou i objekty s jemnou strukturou. Od těchto objektů se zvuk neodráží do širokého okolí, ale jen do úzkého kužele podél osy přicházejícího zvuku, to znamená, že při naklonění objektu směrem od vysílače se odražený zvuk nemusí dostat zpět k senzoru. Při orientaci v prostředí tak reálný dosah senzoru je jen okolo jednoho metru. Pokud by se v místnosti měřilo více ultrasonickými senzory, mohlo by docházet k přerušení spojení nebo k ovlivňování výsledků. Ultrasonický senzor patří do skupiny aktivních senzorů. Pohled na senzor nám přiblíží (**obr. 2.8**).



obr. 2.8 - ultrasonický senzor 1

Zvukový senzor

Zvukový senzor dokáže měřit intenzitu zvuku ve dvou režimech. Buď v režimu, kdy snímá veškerý zvuk přicházející na mikrofon tzv. [dB] režimu nebo v režimu, kdy se zvukový senzor přizpůsobí zvuku, který slyší lidské ucho tzv. [dBA] režimu.

dBA režim - zvukový senzor detekuje decibely v rozsahu slyšitelnosti lidského ucha a rovnoměrně je rozděluje na stupnici od 0 do 100 %.

dB režim - zvukový senzor zachycuje veškerý zvuk přicházející na senzor. Využívá své citlivosti v plném rozsahu. Veškerý přijatý zvuk převádí opět rovnoměrně na stupnici od 0 do 100 %.

Zvukový senzor dokáže měřit tlak zvuku až do úrovně 90 dB. Na (**obr. 2.9**) je pohled na zvukový senzor.



obr. 2.9 - zvukový senzor 1

Světelný senzor

jedná se o další senzor, který dává robotovi zrak (druhý je ultrasonický senzor). Senzor má dvě části vysílací světelnou diodu a přijímací světelnou diodu. Vysílací světelná dioda vysílá červené světlo. Druhá na světlo citlivou diodu, toto světlo přijímá a vyhodnocuje, v jaké intenzitě světlo dopadlo na přijímací diodu. Senzor může pracovat i jinak. Vysílací dioda je vypnuta a na světlo citlivá dioda snímá intenzitu světla z okolního prostředí. Senzor dokáže rozlišit mezi světlem a tmou. Jednotlivé barevné odstíny převádí na stupně šedi a zobrazuje je na stupnici od 0 do 100 %. (**obr. 2.10**).



obr. 2.10 - světelný senzor 1

Dotykový senzor

Dotykový senzor má dvě polohy a to polohu 1, (senzor je stlačen) a polohu 0 (senzor zůstává v klidové poloze). Senzor má tři stavy, ve kterých dává na výstup logickou hodnotu 1. Senzor reaguje buď na stisk tlačítka nebo na uvolnění tlačítka nebo na naražení do tlačítka. Pohled na senzor nám přiblíží (**obr. 2.11**).



obr. 2.11 - dotykový senzor 1

Rotační senzory

V každém z 3. motorů je vestavěn rotační senzor. Tento rotační senzor měří otočení hřídele buď ve stupních v rozsahu 0 - 380° nebo zaznamenává počet otáček rotoru.

2.3.2 Výstupní periferie

Do výstupních periférií se zahrnují motory a světelné diody. Tyto periferie dávají robotovi možnost se pohybovat nebo upozorňovat.

Motory

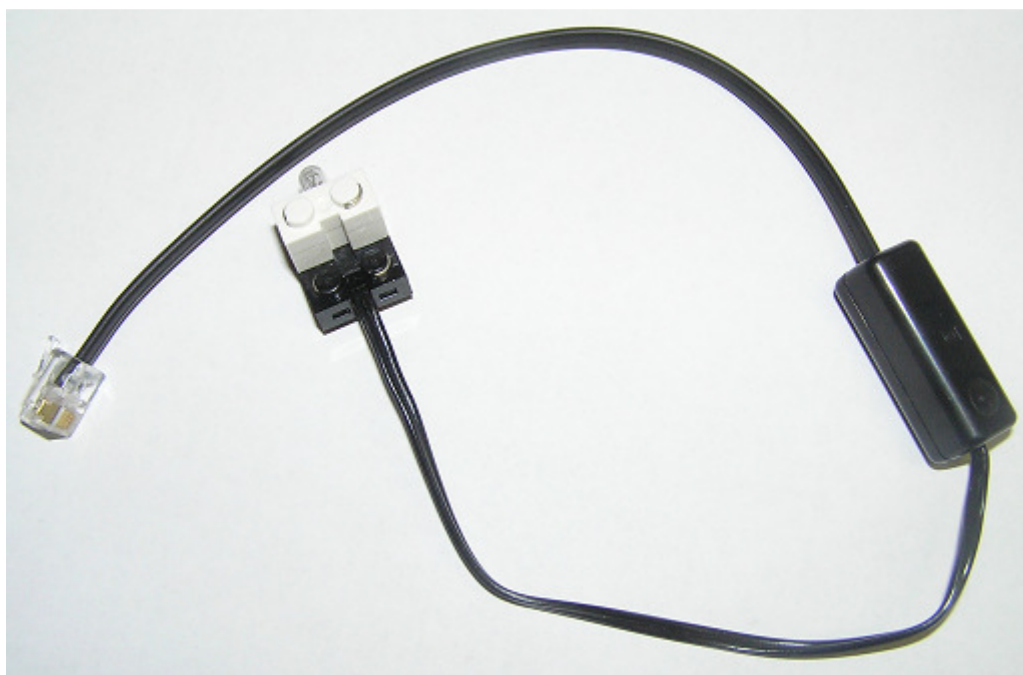
Robot LEGO MINDSTORMS NXT je vybaven třemi servomotory. Pokud jsou motory připojeny na porty A a B, jsou oba motory synchronizovány. Port C není s ostatními porty synchronizován. To nám zaručuje, že při chodu motorů na portech A a B se motory navzájem nepředbíhají. Všechny motory jsou převodovány a obsahují rotační snímače. Na otočné hřídeli jsou po obou stranách ramena v kruhu umístěné 4 čepy. U všech motorů se dá měnit směr otáčení hřídele. Rychlost otáčení je dána aktuálním výkonem dodávaným do motoru přes jednotlivé porty. Rychlost otáčení hřídele motoru se dá nastavit i pomocí rotačního senzoru. Každý motor má dvě připojovací pole, přes něž se dají motory robustně upevnit na nosnou konstrukci. Podobu motorů upřesňuje následující (**obr. 2.12**).



obr. 2.12 - motor 1

Světelné diody

Robot obsahuje celkově 3 světelné LED diody. Každá z těchto diod je zapuštěna do LEGA kostičky o rozměrech 2x2 spojovacích LEGO bodů opatřených napájecími kontakty. Kostičky jsou napájeny přes kabel s redukcí a je možné je připojit na jakýkoliv výstupní port stejně tak, jako jsou připojeny motory. Světelné diody mají jednotnou bílou barvu, ale pokud chceme tuto barvu měnit, můžeme na kostičku umístit barevný kryt. Viz. (**obr. 2.13**).



obr. 2.13 -kabel s redukcí a LED diodou 1

2.3.3 Doplnkové díly

Doplnkové díly tvoří nedílnou součást každého robota. S těmito díly dostává robot nové schopnosti: vnímání, pohyb atd. Pro robota LEGO MINDSTORMS NXT existuje poměrně široká paleta různých doplňků, především senzorů. Pro příklad je uvedeno několik druhů senzorů a jiných částí.:

Kompas senzor

Senzor je digitální a měří magnetické pole země. Na výstup posílá odchylku od daného směru ve stupních s přesností na 1°.

Akcelerometr

Senzor měří ve třech osách zrychlení od -2g do +2g a je schopen také měřit náklon podél těchto os.

Gyroskopický senzor

Obsahuje jednu osu, s níž dokáže měřit rotaci a jako výstupní hodnotu ukazuje úhel rotace za vteřinu rotace.

Existují i další díly (senzor barvy, deska s možností si vytvořit vlastní senzor atd.). Všechny senzory vyrobené pro NXT obsahují I^2C komunikační protokol.

2.4 LEGO MINDSTORMS Education NXT

Software je dodávaný k robotu LEGO MINDSTORMS NXT. Jedná se o grafický programovací jazyk, který je založen na platformě LabVIEW. Software je jednoduchý, intuitivní a umožňuje grafické drag and drop programování. Jednotlivé grafické bloky reprezentují různé funkce částí robota anebo programové části. Program funguje jak pod systémem Microsoft Windows, tak i pod systémem Apple Macintosh. Samotné programování se děje přetahováním grafických ikon na plochu, kde se automaticky spojují do linie a tvoří tak základ funkčního programu jakýsi blokový diagram. Softwarem je možno ovládat i komponenty starší verze Lega RCX. V následující kapitole si popíšeme jednotlivé funkce softwaru.

Samotné prostředí se rozděluje na čtyři části:

1. Levá horní část programového prostředí - umožňuje sestavovat graficky program
2. Pravá horní část – informační, popisuje základní možnosti práce s robotem
3. Levá spodní část - informuje o vybraném bloku a dává nám možnost tento blok dodefinovat

4. Pravá spodní část - odkazuje na nápovědu.

Tyto části společně s průvodcem dávají uživatelsky příznivé programátorské prostředí.

2.4.1 Programová část

Jednotlivé funkční bloky jsou umístěny do levé postranní lišty, kde si bloky lze vybírat ze záložek. Na pravé straně je vymezen prostor pro tvorbu programu s grafickým ovládáním funkce programu a startovací částí.

Postranní lišta

Postranní lišta obsahuje jednotlivé funkční bloky, které jsou členěny do tří sekcí podle typu funkce, jakou zastávají.

1. sekce (**obr. 2.14a**) - jsou zde bloky, které umožňují základní práci s robotem jako je například: pohyb motoru, ovládání displeje, zvuk, ale také i různé smyčky a časovače pro ovládání základní logiky robota. Sekce umožňuje zrychlený přístup k základním vlastnostem robota.

2. sekce (**obr. 2.14b**) – tvoří ji šest záložek pro pokročilejší ovládání robota.



obr. 2.14a - 1. sekce 1



obr. 2.14b - 2. sekce 1



obr. 2.14c - 3. sekce 1

1. záložka (**obr. 2.15**) - obsahuje všechny komponenty, které byly v první sekci.



obr. 2.15 - běžně používané bloky 1

2. záložka (**obr. 2.16**) - tvoří ji funkční bloky umožňující nějakou akci: motor, reproduktor, displej, posílání zpráv přes bluetooth komunikaci, diody a ovládání motoru pro starší verzi LEGA RCX.



obr. 2.16 - akční bloky 1

3. záložka (**obr. 2.17**) - obsahuje všechny senzory (dotykový, zvukový, světelný, rotační a ultrasonický), všechny senzory pro kostku RCX (dotykový, rotační, světelný, a senzor teploty), tlačítka, časovače a záznam zpráv posílaných pře bluetooth.



obr. 2.17 - senzory 1

4. záložka (**obr. 2.18**) - obsahuje bloky pro smysl toku programu jako je blok Wait, který čeká až se na snímači nebo na časovačích stane nějaká akce. Záložka také obsahuje bloky loop, switch a blok stop.



obr. 2.18 - tok programu 1

5. záložka (**obr. 19**) - obsahuje bloky určené pro práci s daty jako je blok logika s možností výběru logický funkcí (AND, OR, XOR a not), blok MATH (sčítání, odčítání, násobení a dělení), porovnávací blok, blok s možností volby rozsahu, blok náhodného čísla v daném rozsahu a blok variabilní proměnné (textové, číselné nebo logické).



obr. 2.19 - práce s daty 1

6. záložka (**obr. 2.20**) - obsahuje vyspělé bloky umožňující práci s textem, práci se soubory, kalibraci senzorů, resetování motoru, blok umožňující navolit, jak dlouho bude kostka NXT v činném stavu.



obr. 2.20 - vyspělé bloky 1

3. sekce (**obr. 2.14a**) - jsou zde bloky my blocks a web download sloužící pro vytvoření nebo pro stáhnutí Individuálního bloku do programu, jenž by mohl sloužit např. pro nové senzory.

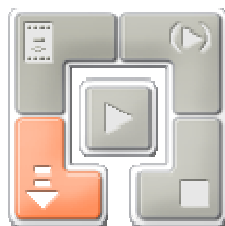
Programovací plocha

Na této ploše se fyzicky vytváří program. Při vytvoření nového programu se na ploše objeví informativní začátek programu (**obr. 2.21**).



obr. 2.21 -informativní začátek programu 1

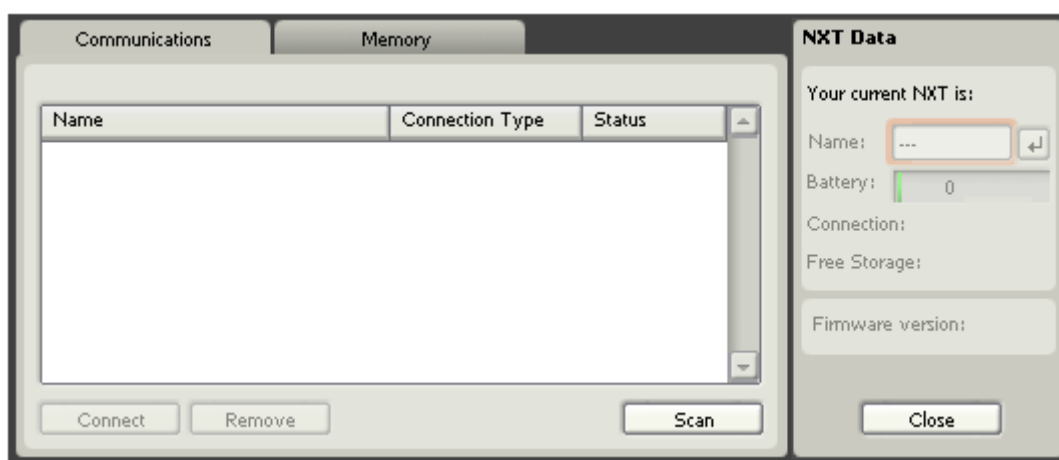
Za tento začátek se postupně skládají jednotlivé bloky tak, aby tvořily funkční program. V pravé dolní části programovací plochy je grafické rozhraní umožňující práci s programem (**obr. 2.22**).



obr. 2.22 - grafické ovládací rozhraní 1

Grafické rozhraní umožňuje:

1. stáhnout program do kostky
2. stáhnout a spustit program
3. stáhnout a spustit vybraný program
4. zastavit běžící program
5. získat informace o kostce (**obr. 2.23**). (Jsou zde informace o typu připojení, jménu kostky, úroveň nabití baterií, verze firmwaru a informace o paměti).



obr. 2.23 - informace o kostce 1

Informační část

V této části jsou ukázky možností činnosti robota LEGO MINDSTORMS NXT. Tyto ukázky jsou v podobě buď několika po sobě jdoucích obrázků, nebo v podobě krátkých videí s ukázkou pohybu robota. Jednotlivé ukázky jsou tematicky rozděleny do skupin podle funkčnosti dané části robota. Je zde k vidění funkční pohyb motorů, činnost jednotlivých senzorů, funkčnost displeje a návrhy sestavení robota.

2.4.2 Robotc

Programovací jazyk založený na standardním programovacím jazyku C, který je určen pro programování robotů LEGO na platformách NXT nebo RCX. Program je alternativou k programovacímu jazyku LEGO MINDSTORMS Education NXT, avšak nejedná se o grafický programovací jazyk, nýbrž o jazyk založený na klasickém textovém programování. Robotc je řízen tokem řádků a není proto řízen tokem dat, jako tomu je u LabVIEW.

Program se skládá ze stejných struktur, jako programovací jazyk C jako jsou např. while, if, for atd. Komponenty a periferie robota mají v programu přiřazeno jednoznačné jméno např. motor[motorA], Sensorvalue(lightSensor) atd.

Příklad programu v robotc.

```
task main()

{

    eraseDisplay();                //vymazání displeje

    nxtDisplayTextLine(4, "  nazdar  "); //zobrazí text na displeji kostky

    motor[motorC] = 100;           //roztočí motor připojený na port C

    wait1Msec(3000);               //zpoždění 3000 ms

    PlayTone( 750, 5);             //zvuk

}
```

Program je velice jednoduchý a ukazuje základní ovládání funkcí robota. Pro navržení složitější aplikace, která by ukazovala modelové řízení a pohyb robota, by aplikace musela mít několik set řádků, což tento program činí nepřehledným a zdlouhavým. Nevýhodou programu robotc je, že při nahrávání do paměti robota potřebuje modifikovaný firmware. LabVIEW toolkit i Robotc při nahrávání programu do robota potřebují rozdílný firmware a nahrávání programu do robota s jiným firmware je nemožné.

3 LabVIEW

Jedná se o grafické vývojové prostředí pro měření a automatizaci založené na programovacím jazyku G. První verze byla vydána v roce 1986 jako nástroj pro programování měřících přístrojů. V průběhu dvaceti let se z LabVIEW postupně stal plnohodnotný software pro návrh, měření a řízení technologických aplikací. LabVIEW umožňuje programování paralelních procesů a běh programu v reálném čase. Dnes se toto paralelní zpracování využívá v počítačích s více jádry a u FPGA polí. Samostatně označení LabVIEW znamená Laboratory Virtual Instrument Engineering Workbench volně přeloženo (laboratorní prostředí pro vývoj virtuálních přístrojů). Původně bylo zaměřeno na automatické měření, ale postupem času se LabVIEW rozšířilo do mnoha oblastí lidské činnosti. Program má stejné programovací struktury jako textové programové prostředí. Od verze 8.20 je LabVIEW rozšířeno o objektové programování a uživatelé zde mohou používat třídy, objekty, metody, zapouzdření atd. Z tohoto grafického prostředí lze volat kód z textových jazyků pomocí převedení kódu do dynamické knihovny DLL nebo sdílené knihovny. Grafický kód z LabVIEW je přímo kompilován do strojového kódu což umožňuje větší rychlost nežli u ostatních programovacích jazyků.

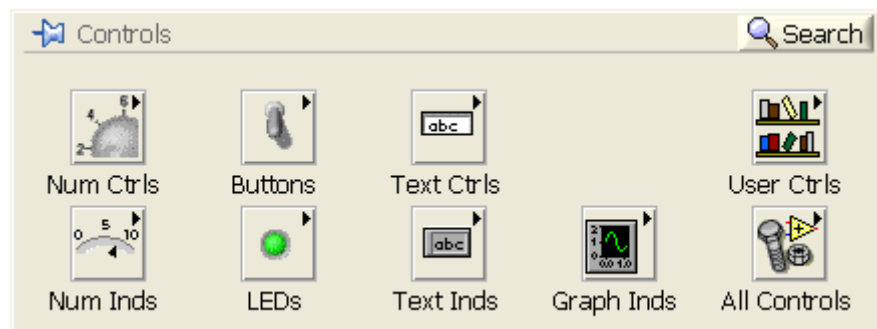
V práci je použit LabVIEW 7.1. Pro programování robota LEGO MINDSTORMS NXT je nutné mít v počítači nainstalovaný LabVIEW toolkit, který umožňuje přenášet a zkompilovat program nahrávaný do paměti kostky NXT. Tento toolkit obsahuje bloky pouze pro robota a bloky, jež jsou universální pro celou platformu LabVIEW a jsou umístěny na blokovém diagramu ve všech funkcích. Následující popis LabVIEW je popisován i s bloky LabVIEW toolkit tzn., které lze použít pro programování robota.

3.1 Popis prostředí LabVIEW

Při spuštění programu si můžeme vybrat z nového programu (VI-Visual instrument) nebo z předdefinovaných kusů programů, jako je předdefinovaný read and display, control template aj. Celé grafické prostředí se rozděluje na dvě části, Front Panel a Block diagram. Část Block diagram umožňuje grafické programování, kde se jednotlivé bloky funkcí a jejich vlastnosti spojují tak, aby vznikl program. Při spuštění aplikace uživatel tuto část nemůže ovlivňovat, ale může ji analyzovat. Naopak v části Front Panel jsou kontrolní prvky a uživatel při spuštění programu může interaktivně tyto prvky kontrolovat, potažmo cílovou funkci programu.

3.1.1 Front Panel

Front Panel je jedna z částí grafického prostředí. Na tuto plochu se umisťují kontrolní prvky, které se vybírají z menu skryté pod pravým tlačítkem (**obr. 3.1**).



obr. 3.1 - kontrolní prvky na Front Panelu 1

Popis jednotlivých částí menu pod pravým tlačítkem.

Num Ctrls – číslíkové řízení. V této záložce jsou bloky, jež umožňují řídit velikost signálu kliknutím na grafickou ikonu na její stupnici nebo vypsání číselné velikosti do ikony. Do této skupiny patří např. knob (otočný knoflík), číselný ukazatel, vertikální a horizontální posuvné ukazatele atd....

Num Inds – číslíkové ukazatele. Záložka obsahuje ukazatele velikosti signálu, jako jsou tank, posuvné ukazatele, metr, číslíkový ukazatel atd....

Buttons – tlačítka. Jsou zde různé druhy tlačítek od pákových až po posuvná.

LEDs – světelné indikátory. V záložce si buď můžete vybrat z čtvercové nebo z kruhové světelné diody.

Text Ctrls – textové prvky. Obsahuje textové konstanty.

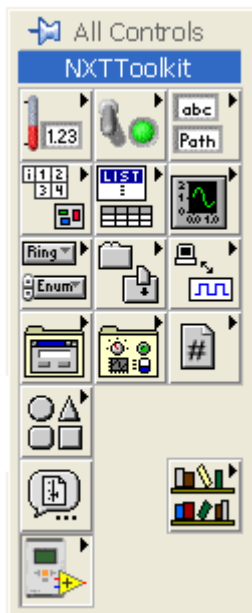
Text Ind – textové ukazatele. V záložce je textová konstanta, textové pole a ukazatel cesty.

Graph inds – grafy. Tato záložka obsahuje různé druhy grafů.

User Ctrls – záložka slouží pro případné bloky, které sám uživatel vytvoří.

Tyto záložky (menu) tvoří zrychlený přístup k nejpoužívanějším blokům.

All Controls (obr. 3.2) - sdružuje všechny výše uvedené skupiny bloků a bloky, jež tvoří dekoraci, speciální bloky pro práci s programem, jako jsou pole, klastry a ostatní bloky, vstupy atd. Při nainstalovaném toolkitu (**obr. 3.3**) záložka All Controls obsahuje ještě bloky, jež jsou součástí daného toolkitu.



obr. 3.2 - menu All controls 1

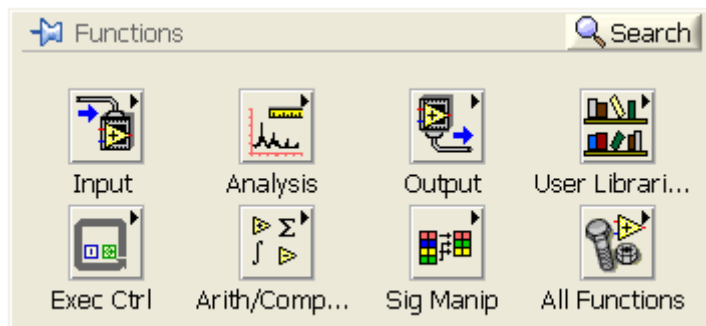


obr. 3.3 - submenu NXT Toolkit 1

Nainstalovaný LabVIEW toolkit obsahuje bloky, jenž jsou stejné jako v knihovnách LabVIEW. Obsahuje však i bloky, jež jsou specifické pro daný toolkit. LabVIEW Toolkit for LEGO MINDSTORMS NXT má například stejné numerické, grafické bloky avšak v jednotlivých submenu se liší v možném počtu jednotlivých bloků nebo jednotlivá menu, jež jsou standardně v LabVIEW, úplně chybí. V části Front Panel má tento LabVIEW toolkit jen nepatrně rozšířenou nabídku rozdílných bloků. V podstatě se jedná o bloky tvořící dekoraci a některé kontrolky.

3.1.2 Block Diagram

V této části vývojového prostředí se sestavuje fyzicky grafický program pomocí spojování bloků funkcí. Bloky funkcí jsou umístěny na ploše pod pravým tlačítkem myši, viz (**obr. 3.4**).



obr. 3.4 - functions na Block Diagramu 1

Popis jednotlivých částí menu pod pravým tlačítkem.

Input – v záložce jsou umístěny bloky, jež mohou měřit vstupní signál a dále ho distribuovat programu, nebo vstupy, jež nám dává virtuální vstupní hodnotu.

Exec Ctrl – obsahuje základní smyčky a bloky s časovou strukturou.

Analysis – záložka obsahuje bloky pro analyzování signálů.

Arith/Comp – v záložce se nacházejí bloky pro numerické operace, booleanovské operace, nebo porovnávací bloky

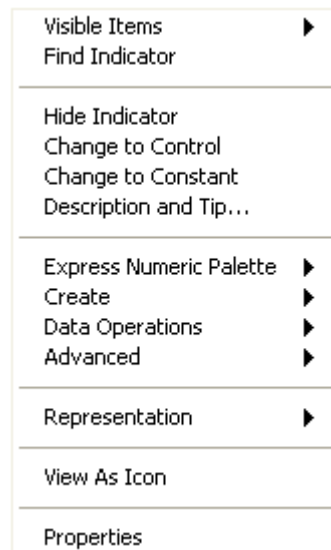
Output – bloky této záložky obsluhují všechny výstupy, jako jsou různé drivery, výstupy na displej, zpracovávají vytvořená data v LabVIEW atd.

Sig Manip – manipulace se signálem. Jsou zde bloky pro slučování, vybírání mezi signálem nebo bloky pro manipulaci se signálem.

All Functions - záložka sdružující všechny bloky na paletě (**obr. 3.5**). V této záložce jsou bloky pro logiku grafického programování a bloky, jež nejsou v paletě pro rychlý přístup, popisovaný výše. Jsou zde smyčky, různé typy proměnných, bloky pro manipulaci s daty jako jsou klastry, pole a podobně, časovače, síťové komunikace a mnoho dalších bloků.



obr. 3.5 - menu All functions 1



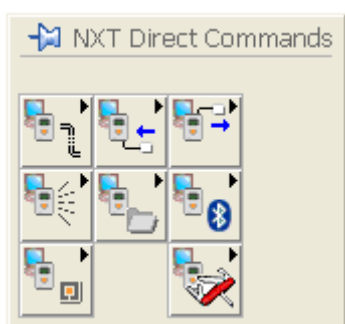
obr. 3.6 - vlastnosti bloku 1

Jednotlivé bloky v LabVIEW se spojují pomocí drátů (spojů). Každý blok v LabVIEW zastává jinou funkci a pracuje s jinými daty. Spoje jsou proto v LabVIEW vyobrazeny v různém provedení a uživatel tak má přehled o tom, jakého typu je vstup nebo výstup daného bloku. Spoje jsou typu string, numeric, Boolean, array, cluster atd. Některé bloky jako je numeric indicator mohou ukazovat údaj i v jiném datovém formátu nežli je na jeho vstup přiváděn jako je formát byte, word, long, extended, double atd.

Vlastnosti bloků (**obr. 3.6**) jsou nastavitelné a jsou zpracované pod pravým tlačítkem myši. V tomto pop up menu jsou vlastnosti jednotlivého bloku a vlastnosti, jež jsou společné všem blokům. Jsou zde položky zahrnující zviditelnění popisku, změnu a vytvoření na konstantu, indikátor nebo na kontrolní prvek, reprezentaci datového formátu, zobrazení palety s podobnými bloky nebo vlastnosti určeného bloku. Každý blok je v LabVIEW prezentován grafickou ikonou, která se dá v tomto pop up menu měnit.

3.2 LabVIEW toolkit for LEGO MINDSTORMS NXT

Toolkit obsahuje základní bloky pro tvorbu programu pro řízení robota LEGO MINDSTORMS NXT. Samotné bloky se prioritně rozdělují do dvou submenu s vlastnostmi přímého řízení (**obr. 3.7**) nebo vložení programu do paměti robota a řízení robota jako autonomního systému (**obr. 3.8**). Obě dvě palety se nacházejí v menu All Function v části Block Diagram.



obr. 3.7 - přímé řízení robota 1



obr. 3.8 - NXT Toolkit 1

3.2.1 NXT Direct Commands

Menu obsahuje bloky pro přímé řízení robota přes počítač. Bloky jsou rozděleny do několika záložek podle funkčnosti jednotlivých komponent.

Connection – záložka obsahuje bloky pro obsluhu komunikace mezi robotem LEGO MINDSTORMS NXT a počítačem. Bloky nám vytvoří komunikaci mezi robotem a počítačem. Zahrnují bloky pro vyhledání robota, propojení se s robotem a vytvoření NXTObject a informace o připojení (bluetooth adresa, typ připojení, jméno robota).

Input – záložka obsahuje bloky jenž obsluhují informace přicházející na vstupní porty. Např. informace od senzorů, obsluha nízkorychlostního digitálního protokolu.

Output – obsahuje bloky pro obsluhu motoru.

Sound – obsahuje bloky pro obsluhu zvuku např. přehrání zvukového souboru, přehrání tónu a zastavení reproduktoru.

File I/O – tato záložka obsahuje bloky pro práci se soubory např. stahování souboru z počítače i kostky NXT, smazání souboru nebo defragmentaci souborů uložených v kostce NXT.

BT Messages – umožňuje číst a zapisovat zprávy posílané prostřednictvím Bluetooth.

Program Execution – bloky této záložky vykonávají řízení programu např. bloky pro zastavení, spuštění a získání jména programu.

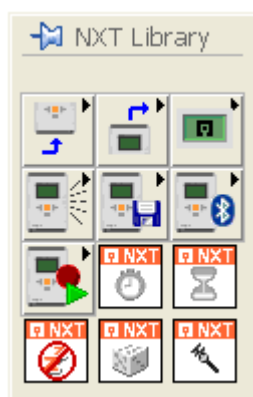
Utilities – tato záložka obsahuje bloky, které nám dávají informace o samotné kostce NXT.

Např. úroveň baterií, určení doby, po kterou je kostka aktivní, verzi firmwaru, získání informací o připojení atd.

3.2.2 NXT Toolkit

Menu obsahuje základní struktury pro tvorbu programu. Tento toolkit obsahuje zredukovanou sadu bloků pro obsluhu programu ve srovnání z LabVIEW. Všechny bloky v tomto menu jakožto i bloky obsažené v menu NXT Direct Command jsou navrženy přímo pro firmware nainstalovaný v kostce NXT a podporují tento toolkit. Některé bloky jsou zredukovány do základního tvaru a neumožňují plného využití potenciálu tohoto bloku. Jako příklad lze uvést blok pro generování n-dimenzionálního pole (build-Array). V tomto bloku lze vytvořit n-dimenzionální pole, avšak firmware v kostce toto n-dimenzionální pole nepodporuje a tudíž se ztrácí podstatná schopnost tohoto bloku. Menu dále obsahuje bloky pro jednoduché ovládání všech komponent robota (**obr. 3.8**).

Program lze vytvořit pomocí standardních prvků, jako jsou pole, různé druhy konstant, klastry, časovače, aritmetické operace a další. Menu obsahuje komponenty pro ovládání robota uložené pod záložkou NXT Library (**obr. 3.9**).



obr. 3.9 - NXT Library 1

Knihovna (NXT Library) obsahuje bloky pro práci se všemi částmi robota. Obsahuje záložky pro práci se vstupy, výstupy, displejem, zvukem, práci se soubory, bluetooth komunikací, záznamem událostí a dalšími.

4 Praktická část

4.1 Cíle praktické části práce

Praktická část bakalářské práce se bude zabývat prozkoumáním možnosti robota LEGO MINDSTORMS NXT. Praktická část se bude zabývat konstrukcí robota, programovým vybavením. Práce bude porovnávat programové prostředí LEGO MINDSTORMS Education NXT s grafickým vývojovým prostředím LabVIEW. Možnosti programového prostředí LabVIEW pro návrh řízení robota. Bude hodnotit výsledky dosažené při návrhu, programování a odladění robota a možnost robota jako celku být modelem pro průmyslové aplikace.

4.2 Konstrukce robota

Robota LEGO MINDSTORMS NXT je možné zkonstruovat do nepřeberných množství tvarů od krácejících robotů až po jezdící. Při konstrukci robota byla největším problémem nedostatečná tuhost nosné konstrukce. Řídící jednotka je lehká, avšak při přidání motorů, senzorů a započtení váhy samotné nosné konstrukce se zátěž na jednotlivé nosníky stala neúměrně vysokou a spojování LEGO kostiček přes styčné body bylo neefektivní. Konstrukce byla tedy volena z kombinace nosných dílů LEGO spojených osami a spoji TECHNICs v minimálně dvoubodovém spojení těchto částí.

Pro úlohu byl vybrán model, který má dvě kola poháněná motorem a jedno volné otočné kolo (**Příloha B**). Pro každou úlohu byla konstrukce robota částečně pozměněna, protože každá úloha využívá jiné senzory nebo jiné akční členy. Obě dvě poháněná kola jsou synchronizována, pokud jsou připojena na výstupní port A, B.

4.3 Zadání úlohy

Návrh a odzkoušení demonstračního programu pro modelové odzkoušení možnosti řízení robota v prostředí LabVIEW.

Řídící jednotka - kostka NXT má 4 vstupy a 3 výstupy. Na vstupy se tedy mohou připojit maximálně 4 senzory a na výstup maximálně 3 jakékoliv akční prvky. Díky této konfiguraci nelze vytvořit vhodnou úlohu, která by zahrnovala celý rozsah všech periferních prvků připojovaných ke kostce NXT. Proto pro modelové odzkoušení řízení robota bylo vytvořeno několik programů zachycujících funkce jednotlivých periférií při modelových situacích pohybu a řízení robota.

4.3.1 Objíždění překážky

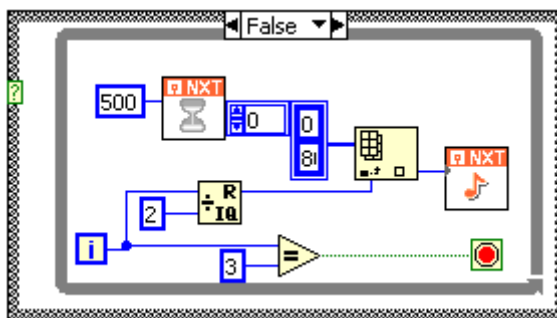
Úloha má za cíl prozkoumat možnosti robota, jeho pohyb podle stanovených instrukcí, synchronizaci obou motorů, funkci displeje jako zobrazovací jednotky aktuálních parametrů, činnost světelného senzoru a součinnost všech těchto komponent v reálném času.

Princip funkce

Robot při zjištění překážky tuto překážku objede podle stanovené cesty. Při svém pohybu částečně využívá již projetou cestu. Při pohybu vzad vydává charakteristický zvuk a ve všech pohybech na displej vypisuje aktuální směr řízení.

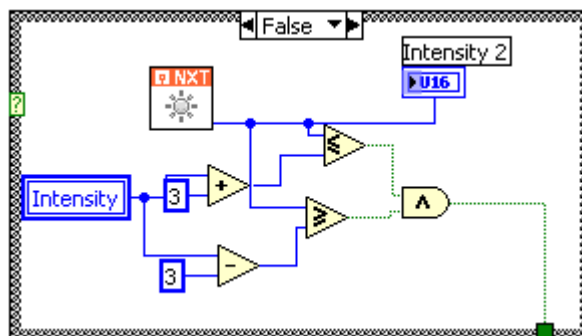
Programové řešení

Robot má ve své paměti uložen rozsah intenzity, na kterou bude reagovat. Jedná se o intenzitu, která je mimo rozsah intenzity okolního prostředí (**obr. 4.1**).



obr. 4.1 - přerušovaný pískot 1

Narazí-li robot na překážku, zastaví a podle nastavení programu nejdříve nastaví zpětný chod na určitý úhel otočení motoru. Následně se otočí, jede dopředu, otočí se zpět do původního směru a jede následně dopředu. Celý cyklus se opakuje, dokud se celá překážka neobjede. Při zpětném chodu motorů vydává motor specifický přerušovaný zvuk (**obr. 4.2**) a při jakémkoliv pohybu robota se na displeji kostky NXT objeví směr pohybu (rovně, vpravo, vlevo, dozadu, dopředu, stojím).



obr. 4.2 -rozmezí intenzity světla 1

Výsledek řešení

Robot pracoval podle předem stanovených kritérií. Během pohybu robota docházelo k jevu, kdy robot se při otáčení odchýlil od předpokládaného úhlu. Proměnlivá vzdáleností kola od osy otáčení způsobila přetočení nebo naopak nedotočení celého robota. Program pracoval dle zadání, ale jednotlivé nedostatky byly způsobeny konstrukční nedokonalostí některých komponent (uchycení kola na ose).

4.3.2 regulace motorů v závislosti na vzdálenosti a aktuálním výkonu motorů

Cílem úlohy byla koordinace akčních členů a reakce na podmínky v závislosti na údajích senzorů v reálném čase.

Princip funkce

Robot zjistí-li překážku, vyhodnotí, v jaké vzdálenosti se překážka nachází a podle aktuální rychlosti otáček motorů a aktuální vzdálenosti překážky zvolí budoucí rychlost otáčení motorů a tím svoji budoucí rychlost. Budoucí rychlost může být kladná i záporná.

Programové řešení

Robot se pohybuje libovolnou rychlostí. Ultrazvukový senzor zachytí pohybující se předmět a podle předem stanovených vlastností se začne robot pohybovat. I když ultrazvukový senzor určuje vzdálenost do 255 cm, pro tento program je maximální vzdálenost, kterou program akceptuje, 100 cm. Tato vzdálenost je volena s ohledem na možnou nepřesnost měření, kdy snímač na tuto vzdálenost měří s přesností ± 3 cm, a také na neschopnost ultrasonického senzoru určit vzdálenost překážky, jenž není kolmo k senzoru nebo je příliš tenká, zakřivená. Uživatel nastavuje, jak bude pracovat řízení motorů. Může nastavit, buď aby robot setrval na přibližně konstantní vzdálenosti nebo jen aby se robot vyhnul srážce v jakékoli vzdálenosti

robota od předmětu. Program přiřazuje k 20. úrovní výkonu motoru v ose Y a 10. úrovní v ose X 0-100% budoucího výkonu motoru. Při následném vyhodnocení budoucího výkonu motoru se výkon rozřazuje do 10 úrovní. Díky tomuto rozřazení by docházelo ke skokovému navýšení budoucího výkonu motoru, a proto je do programu vložen blok lineární interpolace (vzorec. 4.1).

$$y = (x - x_0) * \frac{(y_1 - y_0)}{(x_1 - x_0)} + y_0$$

Vzorec 4.1 - lineární interpolace

Blok lineární interpolace je vyobrazen jako samostatný blok. Celý program je přiložen na cd.

4.3.3 Ostatní úlohy

V ostatních úlohách jsou zařazeny programy, jež prověřují funkci robota nebo programové možnosti, avšak nejsou zařazeny do větší struktury jednoho programu. Proto nejsou uvedeny jako jednotlivé programy.

Tlačítko- tento program testuje možnosti tlačítka, a při stisku dojde k aktivaci motorů na konstantní výkon. Úloha má za cíl prozkoumat vlastnosti tlačítka od reakce na stisk až po reakce na stlačení.

Zvuk – Podle intenzity zvuku je nastavován výkon motoru. Motor je zastavován stiskem tlačítka

Čtverec – robot se pohybuje do čtverce. Program má za cíl prozkoumat všechny možnosti řízení motoru.

Radar – K robotu je připojen ultrasonický senzor, který měří vzdálenost překážky v určitých stupních od osy pohybu. Úhel rozsahu je nastaven na 180° a ultrasonický senzor se pohybuje o 90° na každou stranu sekvenčně po 10°. Konstrukce robota se skládá ze tří motorů uchycených na robustní konstrukci. Řešení robota se nepovedlo uskutečnit, protože ultrasonický senzor je připojen přes kabel s plochým tvarem a tento kabel částečně znemožňuje sekvenční otáčení senzoru a tak celou funkci úlohy.

4.3.4 Řešené problémy

V obou příkladech je řešeno několik problémů, které si jsou částečně podobné.

Podstatným problémem pro práce je velikost paměti robota.

Robot má paměť o kapacitě několik desítek kB což umožňuje nahrát do paměti jen kratší programy. Každý blok jako je motor, senzory, je v prostředí LabVIEW toolkit tvořen složitou strukturou jednoduchých bloků. Celá tato struktura pak zabírá určitou velikost. Program, který obsahuje větší počet těchto bloků, může snadno přesáhnout kapacitu paměti robota.

Zpětnovazební smyčka – V prostředí LabVIEW se automaticky generuje, dojde-li k zacyklení toku dat. Robot, potažmo firmware nepodporují zpětnovazební smyčku a tím se podstatně snižuje použitelnost robota. Ve stanovených úlohách je vyřešen tento problém prostřednictvím lokálních proměnných. Výstupní data se přenáší přes lokální proměnnou zpět na vstup a tím docílují zpětné vazby. Určitým problémem je krokování programu. Program vykoná v jednom kroku obsluhu dílčího programu a v druhém, následujícím kroku, vykoná zachycení výstupní hodnoty zpět na vstupu, to znamená, že program bude regulovat výstupní veličinu vždy jen jednou za dva kroky. Tento problém se vyřeší funkcí programu jen jednou za dva kroky běhu programu.

Vícedimenzionální pole – Vícedimenzionální pole je základní prvek programování. Robot nepodporuje vícedimenzionální pole. V programu, je řešen tento hendikep, case strukturou, kdy ke každému stavu case struktury je přiřazeno jednodimenzionální pole s několika prvky.

Robot nepodporuje i jiné struktury. Příkladem je flat struktura, kdy robot nepodporuje vícerámcovou flat strukturu.

Celkově tyto nevýhody robota kladou na programátora úkony, jenž ho donucuje k tvorbě programů, majících menší paměťová nároky, štíhlejší strukturu a vedoucích k zbytečnému plýtvání s bloky. Programy, nepodporující základní struktury, jsou pro uživatele nepřehledné a stávají se zbytečně zdlouhavými pro výpočet řídicí jednotky.

5 LabVIEW x LEGO MINDSTORMS Education NXT

Programovací prostředí LEGO MINDSTORMS Education NXT je založeno na platformě LabVIEW. Obě prostředí si jsou tudíž dosti podobná. LEGO MINDSTORMS Education NXT je přizpůsobeno jen k řízení robota, avšak prostřed LabVIEW je určeno pro měření a řízení technologických aplikací. Pro řízení robota se do prostředí LabVIEW musí instalovat LabVIEW toolkit for LEGO MINDSTORMS NXT s vlastními bloky pro řízení robota.

5.1 LEGO MINDSTORMS Education NXT

Výhody - jednoduché a intuitivní prostředí. Jednotlivé bloky jsou graficky upraveny tak, aby připomínaly činnost, kterou dané bloky mají vykonávat, například blok sound senzor je graficky vyobrazen jako ruční mikrofón. Tímto způsobem jsou upraveny všechny bloky a umožňují tak vytváření programu i pro uživatele (děti), kteří se nikdy nesetkali s klasickým řádkovým programováním a neznají základní struktury programu. Prostředí umožňuje využívat periferie, jež jsou obsaženy v starší verzi RCX. Každému bloku je přiřazena informativní část, kde se uživatel dozví v přehledu vše, co daný blok umožňuje.

Nevýhody – Navzdory dobrému grafickému prostředí má LEGO MINDSTORMS Education NXT několik nedostatků a absencí oproti LabVIEW. Prostředí umožňuje více cyklové programy, avšak pro uživatele se toto prostředí tímto úkonem stává nepřehledné. Při absenci více cyklového programu se podstatně snižuje operativnost robota a robot je odkázán na jednoduché řízení neumožňující rozsáhlejší práci s přijatými daty. Oproti LabVIEW postrádá prostředí prvky, jež pracují s polem, klastry, sekvenční flat struktury, některé matematické a porovnávací funkce atd., což snižuje možnosti řízení robota. Prostředí nemá možnost přímého řízení robota

5.2 LabVIEW

Výhody – toolkit umožňuje přímo řídit robota, to znamená, že nepotřebujeme nahrávat program do paměti a tím se omezovat jejími kapacitními schopnostmi. Všechny struktury jsou rozděleny do několika submenu a tím ulehčují práci s programováním. Graficky každý blok nabízí více možností propojení mezi jednotlivými komponentami oproti LMENXT (LEGO MINDSTORMS Education NXT) a tím zpřehledňuje samotné programování. Navíc přidává funkční bloky pro práci s programem, jako jsou klastry, práce s poli, flat struktura, práce s datovými typy atd. Přidáním bloků se z tohoto testovacího grafického programu stává poměrně účinný nástroj pro řízení robota i jiných průmyslových aplikací. Silnou stránkou oproti LMENXT je možnost grafické analýzy signálů, práce s dekoracemi atd.

Nevýhody – Celý LabVIEW toolkit je nainstalován pod platformou LabVIEW 7.1. Tato verze LabVIEW má nevýhodu, že při větším počtu submenu se výběr bloku může posunout až na několik podúrovní a samotný blok je z pohledu programátora „zahrabaný“. Určitou nevýhodou v přímém řízení robota je nemožnost obsluhovat některé funkce robota jako je práce s displejem. V celku podstatnou věcí při programování robota je, že některé bloky neumožňují využívat svůj potenciál např. pole, zpětnovazební smyčky...

Celkově je pro návrh aplikací pro řízení robota vhodnější LabVIEW toolkit ve srovnání s LMENXT, jelikož poskytuje širší škálu řídicích bloků a práci s jejich vlastnostmi. Lepší práce s daty, vyhodnocovací stupně atd.

6 Závěr

V práci byly otestovány vlastnosti a možnosti řízení robota LEGO. Testování robota bylo prováděno dle jednotlivých programů, kde každý program zvlášť testoval funkčnost dané komponenty a periférií.

Samotné programování provázely nedostatky programového vybavení robota. Software nepodporoval vícedimenzionální pole, zpětnovazební smyčky, vícerázovou flat strukturu, čímž se razantně znehodnocovalo programátorské prostředí. Nepřehlédnutelnou položkou byla i nedostatečná velikost paměti pro nahrávání programu.

Robota jako celek lze použít pro návrh průmyslových aplikací, ale má několik nedostatků, které firma LEGO musí dořešit. Např. nedotažené programové vybavení v programovacím jazyku LEGO MINDSTORMS Education NXT, kde program má nedostatky především v možnostech funkčních bloků. Mohla by se zkvalitnit programové ovládání displeje, vnitřní paměti.

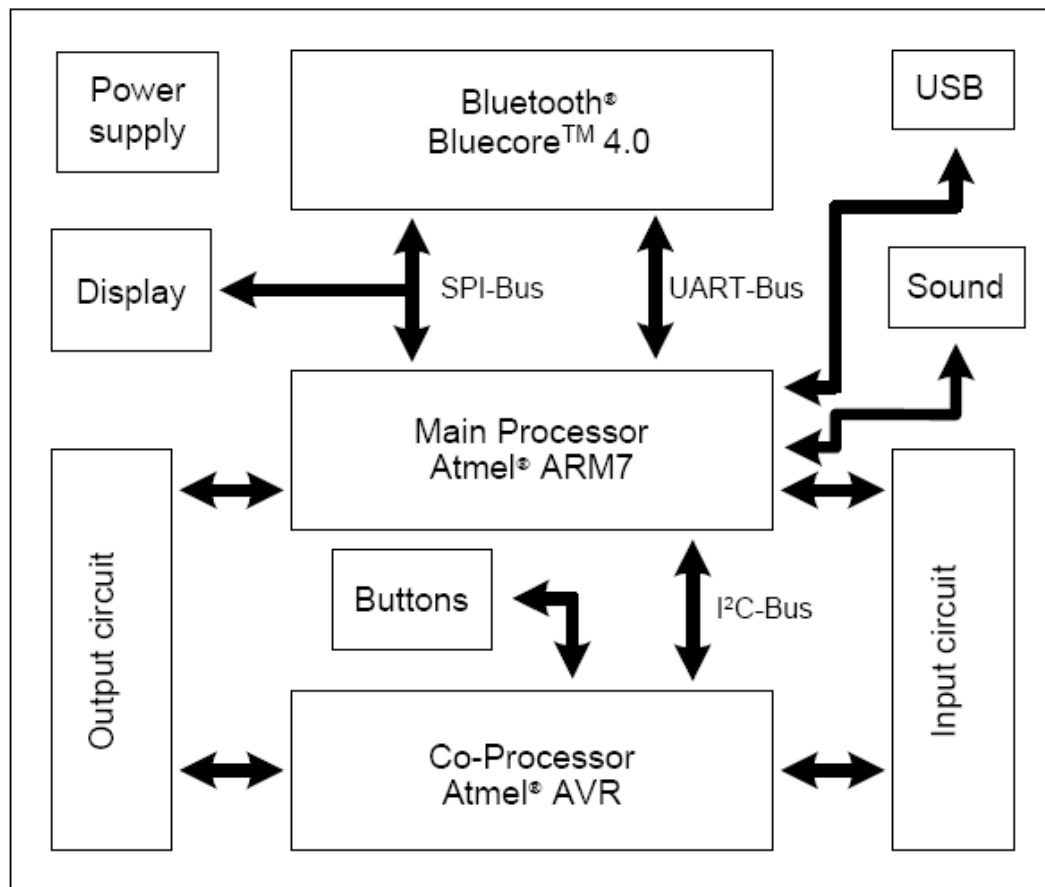
Program by zlepšilo zařazení více struktur pro programování, zejména byl patrný malý výběr druhů smyček. Konstrukci robota lze však považovat za dostatečnou.

Robot se především hodí vzhledem k jeho nízké ceně a lehké konstrukci oproti komerčním průmyslovým robotům, pro návrh průmyslových aplikací. Jeho vlastnosti mohou využívat školy i firmy.

Literatura

- [1.] *Atmel* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://www.atmel.com>>.
- [2.] *AVR* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://avr.hw.cz/>>.
- [3.] *LEGO MINDSTORMS* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://mindstorms.lego.com>>.
- [4.] *National Instruments* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://www.ni.com/>>.
- [5.] *Eduxe* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://www.eduxe.cz>>.
- [6.] *ROBOTC.net* [online]. 2008 [cit. 2008-04-14]. Dostupný z WWW: <<http://www.robotc.net/>>.
- [7.] Více než 20 let inovací LabVIEW. *Automatizace*. říjen 2007, roč. 50, č. 10, s. 673-674.

Příloha A



Příloha B

